```
RRRRRRRRRRR      UUU          UUU  NNN         NNN     000000000     FFFFFFFFFFFFFFF  FFFFFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN         NNN     000000000     FFFFFFFFFFFFFFF  FFFFFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN         NNN     000000000     FFFFFFFFFFFFFFF  FFFFFFFFFFFFFFF
RRR          RRR UUU          UUU  NNN         NNN  000         000  FFF              FFF
RRR          RRR UUU          UUU  NNN         NNN  000         000  FFF              FFF
RRR          RRR UUU          UUU  NNN         NNN  000         000  FFF              FFF
RRR          RRR UUU          UUU  NNNNNN      NNN  000         000  FFF              FFF
RRR          RRR UUU          UUU  NNNNNN      NNN  000         000  FFF              FFF
RRR          RRR UUU          UUU  NNNNNN      NNN  000         000  FFF              FFF
RRRRRRRRRRRR     UUU          UUU  NNN  NNN    NNN  000         000  FFFFFFFFFFFF     FFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN  NNN    NNN  000         000  FFFFFFFFFFFF     FFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN  NNN    NNN  000         000  FFFFFFFFFFFF     FFFFFFFFFFFF
RRR    RRR       UUU          UUU  NNN      NNNNNN  000         000  FFF              FFF
RRR    RRR       UUU          UUU  NNN      NNNNNN  000         000  FFF              FFF
RRR    RRR       UUU          UUU  NNN      NNNNNN  000         000  FFF              FFF
RRR       RRR    UUU          UUU  NNN         NNN  000         000  FFF              FFF
RRR       RRR    UUU          UUU  NNN         NNN  000         000  FFF              FFF
RRR       RRR    UUU          UUU  NNN         NNN  000         000  FFF              FFF
RRR          RRR UUUUUUUUUUUUUUU  NNN         NNN     000000000     FFF              FFF
RRR          RRR UUUUUUUUUUUUUUU  NNN         NNN     000000000     FFF              FFF
RRR          RRR UUUUUUUUUUUUUUU  NNN         NNN     000000000     FFF              FFF
RRR          RRR UUUUUUUUUUUUUUU  NNN         NNN     000000000     FFF              FFF
RRR          RRR UUUUUUUUUUUUUUU  NNN         NNN     000000000     FFF              FFF
```

_$2

```
Sym
---
NDX
NDX
NUM
NUM
OPE
OUT
PAC
PAC
PAC
PAC
PAC
PAC
PAC
PAD
PAG
PAG
PAG
PAG
PAG
PAG
PAG
PER
PUT
RCO
RIN

RLI
RNO
RNO
RTY
SAV
STR
STR
STR
STR
STR
STR
STR

 STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
```

TOC

!
LIT

!

!

!

!

!

!
LITE

!
! SE
LITE

!
! SE
LITE

!

```
Version:      'V04-000'
```



```
++
FACILITY:      DSR (Digital Standard RUNOFF) / DSRPLUS

ABSTRACT:
      Definitions having to do with SCA (SCANT control area).

ENVIRONMENT:   Compatible BLISS

AUTHOR:        Rich Friday

CREATION DATE: 1978

MODIFIED BY:

      016      REM00016        Ray Marshall    16-November-1983
               Added macros with "_f_" within their names and defining cells
                 so they can be loaded by Ken's SAVRES module using macro
                 names instead of numeric offsets.

      015      REM00015        Ray Marshall    7-November-1983
               Added 12 more cells to the case rules section to support case
                 changing within the DEC multinational character set.  This
                 also required renumbering the whole table!

      014      KFA00014        Ken Alden       18-Sep-1983
               For DSRPLUS: added sca_margin_pad.

      013A     KFA00013A       Ken Alden       30-Sep-1983
               Comment changes only; for three items that moved
```

```
!                 from save to save all.
!
!       013     KFA00013        Ken Alden       13-Sep-1983
!               For DSRPLUS: Added SCA_WRD_PASS to the SCA_PASS word [35]
!                 and made SCA_PASS into a bit.
!
!       012     KFA00012        Ken Alden       05-Jul-1983
!               added SCA_FLAGS to the save list.
!
!       011     KFA00011        Ken Alden       15-Mar-1983
!               For DSRPLUS: added SCA[35] SCA_PASS for flags passthrough
!               For DSR:     Extended SAVE & RESTORE capability.
!
!       010     KAD00010        Keith Dawson    07-Mar-1983
!               Global edit of all modules. Updated module names, idents,
!               copyright dates. Changed require files to BLISS library.
!--
```

```
MACRO
        SCA_FC_UT       = SCA[00]%,     !Save case rules here when
        SCA_OC_UT       = SCA[01]%,     !exceptions are set up.
        SCA_FC_LT       = SCA[02]%,     !...
        SCA_OC_LT       = SCA[03]%,     !...
        SCA_WRD_FC_UT   = SCA[04]%,     !Case rules for current word.
        SCA_WRD_FC_LT   = SCA[05]%,     !...
        SCA_WRD_OC_UT   = SCA[06]%,     !...
        SCA_WRD_OC_LT   = SCA[07]%,     !...
        SCA_FCBE_UT     = SCA[08]%,     !Default case rules.
        SCA_OCBE_UT     = SCA[09]%,     !...
        SCA_FCBE_LT     = SCA[10]%,     !...
        SCA_OCBE_LT     = SCA[11]%,     !...
        SCA_MNFC_UT     = SCA[12]%,     !Save case rules here when
        SCA_MNOC_UT     = SCA[13]%,     !exceptions are set up.
        SCA_MNFC_LT     = SCA[14]%,     !...
        SCA_MNOC_LT     = SCA[15]%,     !...
        SCA_MNWRD_FC_UT = SCA[16]%,     !Case rules for current word.
        SCA_MNWRD_FC_LT = SCA[17]%,     !...
        SCA_MNWRD_OC_UT = SCA[18]%,     !...
        SCA_MNWRD_OC_LT = SCA[19]%,     !...
        SCA_MNFCBE_UT   = SCA[20]%,     !Default case rules.
        SCA_MNOCBE_UT   = SCA[21]%,     !...
        SCA_MNFCBE_LT   = SCA[22]%,     !...
        SCA_MNOCBE_LT   = SCA[23]%,     !...
        SCA_WORD_SET    = SCA[24]%,     !TRUE if case rules set for a word only.
!*************************************************************************
!Everything above this point is counted in SCA_CASE_SIZE.

        SCA_JUSTIFY     = .SCA[25]%,    !(SAVE)TRUE if text should be justified.
        SCA_f_JUSTIFY   = SCA[25]%,
        SCA_FILL        = .SCA[26]%,    !(SAVE)TRUE if filling lines.
        SCA_f_FILL      = SCA[26]%,
        SCA_CC_OK       = .SCA[27]%,    !(SAVE ALL)TRUE if control characters allowed in input file
        SCA_f_CC_OK     = SCA[27]%,
        SCA_CROCK       = .SCA[28]%,    !(SAVE)See FJNFNJ for explanation.
        SCA_f_CROCK     = SCA[28]%,
        SCA_LM          = .SCA[29]%,    !(SAVE)The left margin.
        SCA_f_LM        = SCA[29]%,
        SCA_RM          = .SCA[30]%,    !(SAVE)The right margin.
        SCA_f_RM        = SCA[30]%,
        SCA_SPACING     = .SCA[31]%,    !(SAVE)1+number of blank lines between lines of text.
        SCA_f_SPACING   = SCA[31]%,
        SCA_PERIOD      = .SCA[32]%,    !(SAVE ALL)TRUE if double spacing after ".?!;:"
        SCA_f_PERIOD    = SCA[32]%,
        SCA_KER         = .SCA[33]%,    !(SAVE)TRUE if empty records have significance.
        SCA_f_KER       = SCA[33]%,
        SCA_BAR_CHAR    = .SCA[34]%,    !(SAVE ALL)Change bar character to be used if enabled.
        SCA_f_BAR_CHAR  = SCA[34]%,
        SCA_AUTOTITLE   = .SCA[35]%,    !(SAVE)TRUE if .AUTOTITLE is in effect.
        SCA_f_AUTOTITLE = SCA[35]%,
        SCA_FLAGS       = .SCA[36]%,    !TRUE if flags are enabled.
        SCA_f_FLAGS     = SCA[36]%,
!
!***************************  end of save area  ***************************
!
```

```
        SCA_FC          = SCA[37]%,     !TRUE if first character on the line.
        SCA_NBITS       = SCA[38]%,     !SEE BELOW
        SCA_X_FLAG      = SCA[39]%,     !TRUE if in the middle of a sequence
                                        !marked by the <INDEX flag>
        SCA_FRC_CASE    = SCA[40]%,     !TRUE if case of current
                                        !word was forced.
        SCA_CONT        = SCA[41]%,     !TRUE if user said .NO SPACE
        SCA_DO_NBITS    = SCA[42]%,     !SEE BELOW
        SCA_PRESCAN     = SCA[43]%,     !See SCANT for explanation
        SCA_HEADER      = SCA[44]%,     ! Used by FLIP -- True if
                                        ! collecting a header level.
        SCA_SECT_EMPTY  = SCA[45]%,     !TRUE if nothing in current section.
        SCA_XROUTINE    = SCA[46]%,     !Indicates which routine to call for indexing:
                                        !FALSE ==> XR, TRUE ==> SUBXR.
%IF DSRPLUS %THEN
        SCA_PASS        = (SCA[47])<0,1>%,      !TRUE user is passing escape
                                                !sequences.
        SCA_WRD_PASS    = (SCA[47])<1,1>%,      !TRUE user is passing escape
                                                !sequences along with this word.
        SCA_MARGIN_PAD  = (SCA[47])<8,8>%,      !Number of spaces added at the
                                                !beginning of MRA.
%FI
!   ...
        SCA_WRD_NBITS   = SCA[48]%,     !SEE BELOW
        SCA_WRD_CNBITS  = SCA[49]%,     !SEE BELOW
        SCA_WRD_ACNBITS = SCA[50]%,     !SEE BELOW
        SCA_RSKIPS      = SCA[51]%,     !TRUE if multiple spaces/tabs are to
                                        !be skipped.
!   ...
        SCA_FC_CASE     = SCA[52]%,     !TRUE if case rules to be used are
                                        !those for the first character of a word.
        SCA_INDEX       = SCA[53]%,     !TRUE if indexing commands are to be obeyed.
        SCA_FRC_CHR     = SCA[54]%,     !True if current character should not be translated.
        SCA_INDENT      = SCA[55]%,     !Pending indentation.
        SCA_PARA_PND    = SCA[56]%,     !TRUE if a paragraph is pending.
!   ...
!
!       Everything below this point refers to the word currently being
!       scanned.
!*****************************************************************
        SCA_WRD_PNTR    = SCA[62]%,     !A CH$PTR to the first character of the word.
        SCA_WRD_INT_L   = SCA[63]%,     !Internal representation size so far.
        SCA_WRD_EXT_L   = SCA[64]%,     !External size (i.e., print positions)
        SCA_WRD_ISEQN   = SCA[65]%,     !Input line counter or record number.
        SCA_WRD_DRAFT   = SCA[66]%,     !TRUE if word is inside a /DRAFT area.
        SCA_WRD_DRAFT_F = SCA[67]%,     !The draft flag for this word.
        SCA_WRD_BARS    = (SCA[68])<BAR__>%,    !TRUE if change bars associated with this word.
        SCA_WRD_BAR_CHR = SCA[69]%,     !Use this character as the change bar.
        SCA_WRD_CPERD   = SCA[70]%,     !Character being worked on.
        SCA_WRD_SEQN_F  = SCA[71]%,     !TRUE if SCA_WRD_ISEQN is an SOS style record number.
        SCA_WRD_IPAGEN  = SCA[72]%,     !Input page number.
        SCA_WRD_FOOTW   = SCA[73]%,     !The number of footnotes attached to
                                        !this word.
        SCA_WRD_F_XTN   = SCA[74]%,     !First transaction number associated with this word.
        SCA_WRD_L_XTN   = SCA[75]%,     !Last transaction number associated with this word.
!   ...
```

```
        SCA_WRD_LST_HYP = SCA[80]%,     !TRUE if word to end with a '-'.
        SCA_WRD_HYP_PTR = SCA[81]%,     !If SCA_WRD_LST_HYP is TRUE,
                                        !then a CH$PTR to the '-'.
        SCA_WRD_LC_PNCT = SCA[82]%,     !TRUE if last character was end-of-sentence punctuation.
        SCA_WRD_LST_SP  = SCA[83]%,     !Number of spaces after last word.
        SCA_WRD_LST_JUS = SCA[84]%,     !True if justification mark after last word.
        SCA_WRD_LST_UND = SCA[85]%;     !True if last space was underlined.

LITERAL
        SCA_CASE_SIZE   = 25,           !Number of cells containing case information
        SCA_SIZE        = 96;           !Number of cells in entire SCA
LITERAL
        SCA_SAVE_START  = 25,           !Starting number of the SAVEd SCA bits
        SCA_SAVE_END    = 36;           !Ending    "      "  "   "   "   "

!NOTE**** For all fields having to do with underlining, bolding, etc, see ENDCHR, ENDWRD, OUTLIN, and DOFLG
!         to see how these fields get manipulated.

!The bits defined by these macros get set to TRUE if SCANT is supposed
!to BoLD and/or UNDerline all characters it encounters.  These bits get
!set/unset by things like ^& and \&, and .ENABLE/.DISABLE BOLDING, etc.
MACRO
        SCA_BLD         = (SCA_NBITS)<BLD__>%,
        SCA_UND         = (SCA_NBITS)<UND__>%,
        SCA_BLDUND      = (SCA_NBITS)<BLDUND__>%;   ! Bolding and underlining
                                                    ! as a set.

!The bits defined by these macros determine whether or not various flags
!have any effect.  TRUE means that the corresponding function should be
!done when the flag is recognized, FALSE not.
!These bits get set/cleared by commands such as .ENABLE/.DISABLE BOLDING,
!and so on.   See FLGSEM, which sets these flags.
!There is a difference between SCA_XXX and SCA_DO_XXX.  The former
!indicates whether or not the XXX type of emphasis has been turned on
!by a construct such as ^&.  The latter indicates whether or not the
!emphasis called for by XXX should really be done.  In particular,
!consider the following sequence:
!00100 .ENABLE UNDERLINING;^&
!00200 This text will be underlined because both .ENABLE UNDERLINING and _^_&
!00300 were specified.  But
!00400 .DISABLE UNDERLINING;this text (after the ";") will not be underlined
!00500 because the .DISABLE UNDERLINING command indicates it should not be
!00600 done.  Or, perhaps more interesting is the fact that even
!00700 &t&h&e&s&e &t&w&o &w&o&r&d&s will not be underlined whereas
!00800 .ENABLE UNDERLINING;these two words\& will be underlined, but nothing
!00900 after the _\_& sequence.
MACRO                                               !See FLGSEM
        SCA_DO_BLD      = (SCA_DO_NBITS)<BLD__>%,
        SCA_DO_UND      = (SCA_DO_NBITS)<UND__>%,
        SCA_DO_BLDUND   = (SCA_DO_NBITS)<BLDUND__>%,    !Bolding and underlining as a set.
        SCA_DO_OVR      = (SCA_DO_NBITS)<OVR__>%,
        SCA_DO_IND      = (SCA_DO_NBITS)<IND__>%,
        SCA_DO_HYP      = (SCA_DO_NBITS)<HYP__>%;

MACRO
```

```
! The fields defined here are accumulated for an entire word.
! They get cleared at the start of a new word.

        SCA_WRD_BLD     = (SCA_WRD_NBITS)<BLD__>%,
        SCA_WRD_UND     = (SCA_WRD_NBITS)<UND__>%,
        SCA_WRD_BLDUND  = (SCA_WRD_NBITS)<BLDUND__>%,   !Bolding and underlining as a set
        SCA_WRD_OVR     = (SCA_WRD_NBITS)<OVR__>%;

MACRO
! The fields defined here get set just before a new character
! is picked up.  They are inherited from the global environment
! in effect at that time.

        SCA_WRD_C_BLD   = (SCA_WRD_CNBITS)<BLD__>%,
        SCA_WRD_C_UND   = (SCA_WRD_CNBITS)<UND__>%,
        SCA_WRD_C_BLDUN = (SCA_WRD_CNBITS)<BLDUND__>%,  !Bolding and underlining as a set
        SCA_WRD_C_OVR   = (SCA_WRD_CNBITS)<OVR__>%;

MACRO
! The fields defined here get set as various functions are
! requested, on a once-only basis (e.g., single character underline, &x).  These fields, together with
! the previous three fields, determine what functions have been
! requested for a specific character.

        SCA_WRD_AC_BLD  = (SCA_WRD_ACNBITS)<BLD__>%,
        SCA_WRD_AC_UND  = (SCA_WRD_ACNBITS)<UND__>%,
        SCA_WRD_AC_BLUN = (SCA_WRD_ACNBITS)<BLDUND__>%, !Bolding and underlining as a set
        SCA_WRD_AC_OVR  = (SCA_WRD_ACNBITS)<OVR__>%;

MACRO
        SCA_DEFINITION  =
                VECTOR[SCA_SIZE]%;

$FIELD H_R_SCA_FIELDS =
        SET
        H_R_$G_SCA_JUSTIFY      = [$INTEGER],
        H_R_$G_SCA_FILL         = [$INTEGER],
        H_R_$G_SCA_CC_OK        = [$INTEGER],
        H_R_$G_SCA_CROCK        = [$INTEGER],
        H_R_$G_SCA_LM           = [$INTEGER],
        H_R_$G_SCA_RM           = [$INTEGER],
        H_R_$G_SCA_SPACING      = [$INTEGER],
        H_R_$G_SCA_PERIOD       = [$INTEGER],
        H_R_$G_SCA_KER          = [$INTEGER],
        H_R_$G_SCA_BAR_CHAR     = [$INTEGER],
        H_R_$G_SCA_AUTOTITLE    = [$INTEGER],
        H_R_$G_SCA_FLAGS        = [$INTEGER]
        TES;

LITERAL
        H_R_SCA$K_LENGTH = $FIELD_SET_SIZE;

LITERAL
        MAX_H_R_SCA = 3;        !This means there are a maximum of 3 concurrent PUSH_SCAs.
MACRO
   $H_R_SCA_BLOCK =
```

```
        BLOCK [H_R_SCA$K_LENGTH] FIELD (H_R_SCA_FIELDS) %;

MACRO
   PUSH_SCA =
        BEGIN
        PP_SCA [ H_R_$G_SCA_JUSTIFY ]   = .SCA_JUSTIFY;
        PP_SCA [  H_R_$G_SCA_FILL ]     = .SCA_FILL;
        PP_SCA [  H_R_$G_SCA_CC_OK ]    = .SCA_CC_OK;
        PP_SCA [  H_R_$G_SCA_CROCK ]    = .SCA_CROCK;
        PP_SCA [  H_R_$G_SCA_LM ]       = .SCA_LM;
        PP_SCA [  H_R_$G_SCA_RM ]       = .SCA_RM;
        PP_SCA [  H_R_$G_SCA_SPACING ]  = .SCA_SPACING;
        PP_SCA [  H_R_$G_SCA_PERIOD ]   = .SCA_PERIOD;
        PP_SCA [  H_R_$G_SCA_KER ]      = .SCA_KER;
        PP_SCA [  H_R_$G_SCA_BAR_CHAR ] = .SCA_BAR_CHAR;
        PP_SCA [  H_R_$G_SCA_AUTOTITLE ]= .SCA_AUTOTITLE;
        PP_SCA [  H_R_$G_SCA_FLAGS ]    = .SCA_FLAGS
        END
        %;

MACRO
   POP_SCA =
        BEGIN
        SCA_JUSTIFY     = .PP_SCA [  H_R_$G_SCA_JUSTIFY ];
        SCA_FILL        = .PP_SCA [  H_R_$G_SCA_FILL ];
        SCA_CC_OK       = .PP_SCA [  H_R_$G_SCA_CC_OK ];
        SCA_CROCK       = .PP_SCA [  H_R_$G_SCA_CROCK ];
        SCA_LM          = .PP_SCA [  H_R_$G_SCA_LM ];
        SCA_RM          = .PP_SCA [  H_R_$G_SCA_RM ];
        SCA_SPACING     = .PP_SCA [  H_R_$G_SCA_SPACING ];
        SCA_PERIOD      = .PP_SCA [  H_R_$G_SCA_PERIOD ];
        SCA_KER         = .PP_SCA [  H_R_$G_SCA_KER ];
        SCA_BAR_CHAR    = .PP_SCA [  H_R_$G_SCA_BAR_CHAR ];
        SCA_AUTOTITLE   = .PP_SCA [  H_R_$G_SCA_AUTOTITLE ];
        SCA_FLAGS       = .PP_SCA [  H_R_$G_SCA_FLAGS ]
        END
        %;
```